AMENDMENT TO THE CLAIMS

Please **AMEND** claim 20 as follows.

A copy of all pending claims and a status of the claims is provided below.

1. (original) A method of accessing data in a non-relational database, the method comprising the steps of:

creating a master view having a master view index referencing the data;

creating a subordinate view of the master view having a subordinate view index referencing a subset of said master view index, where the subordinate view defines accessible portions of the data and the subordinate view index is linked to a subset of the master view index; and

accessing the data via the subordinate view.

2. (original) The method of claim 1, wherein the creating a master view includes defining at least one of sorted and categorized columns associated with the master view.

3. (original) The method of claim 1, wherein the creating a subordinate view step includes defining at least one of a collapsed subordinate view and a non-collapsed subordinate view.

4. (original) The method of claim 1, further comprising automatically managing the subordinate view.

5. (original) The method of claim 1, wherein the accessing step includes creating an index map which links the accessible data associated with the subordinate view to the master index.

6. (original) The method of claim 5, further comprising caching at least one of the subordinate view and the temporary index map, wherein the caching step includes:

checking whether a predetermined time period has elapsed by checking an elapsed time period counter;

if elapsed, checking whether access frequency exceeds a predetermined threshold by checking an access counter for the subordinate view; and

if the predetermined threshold is exceeded, checking whether the at least one of the subordinate view and the index map can be cached,

if so, then caching at least one of the subordinate view and the temporary index.

7. (original) The method of claim 6, further comprising resetting one of the elapsed time period counter to start a new elapsed time period for counting access frequencies and the access counter for counting access frequencies to the temporary index map during the new elapsed time period.

8. (original) The method of claim 1, further comprising maintaining historical information including access frequency to the subordinate view.

9. (original) The method of claim 1, wherein the accessing step provides one of a

3

reduction of data transferred to a client in a client-server architecture, a decrease in the amount of data manipulated during the accessing step, a decrease in response time to an access request, an increased performance, and a decrease in index size.

10. (original) The method of claim 1, wherein the data includes at least one of categorized non-hierarchical data., hierarchical data, and categorized hierarchical data.

11. (original) The method of claim 1, wherein the master view has a master index referencing at least a portion of the hierarchical data in the non-relational database.

12. (original) The method of claim 1, wherein the creating a subordinate view step includes creating a plurality of subordinate views associated with one or more master views.

13. (original) A method of enhancing performance when accessing hierarchical data in a non-relational database, the method comprising the steps of:

creating at least one subordinate view having a subordinate index referencing a subset of a master index of at least one master view;

creating a subordinate view index map associated with the at least one subordinate view when accessing the hierarchical data; and

accessing the at least a portion of hierarchical data by using the subordinate view index map,

4

wherein an amount of data accessed using the at least one subordinate view is less than the amount of data when accessing the at least one master view.

14. (original) The method of claim 13, wherein the creating at least one subordinate view includes defining at least one of sorted and categorized columns associated with the at least one master view.

15. (original) The method of claim 13, further comprising maintaining historical information including access frequency to the subordinate view.

16. (original) The method of claim 15, further comprising caching at least one of the at least one subordinate view, the subordinate view index map and temporary index, wherein the caching step includes:

checking whether a predetermined time period has elapsed by checking an elapsed time period counter; and

if elapsed, checking whether the access frequency exceeds a predetermined threshold by checking a access counter for the at least one subordinate view;

if the predetermined threshold is exceeded, checking whether the at least one subordinate view can be cached,

if so, caching at least one of the at least one subordinate view, the subordinate view index map, and the temporary index.

17. (original) The method of claim 13, further including assigning priorities to one of the

at least one master view and at least one subordinate view to grade performance.

18. (original) The method of claim 13, wherein the subordinate view includes at least one of a collapsed view and a non-collapsed view and access via the collapsed view providing less data than access via a non-collapsed view.

19. (original) An apparatus for accessing hierarchical data in a non-relational database, the apparatus comprising:

a first component to create a master view having a master view index referencing hierarchical data;

a second component to create a subordinate view of the master view that has a subordinate view index that references a subset of said master view index, the subordinate view defines accessible data of the hierarchical data and the subordinate view index linked to a subset of the master view index; and

a third component to access the hierarchical data via the subordinate view.

20. (currently amended) A computer program product comprising a computer usable medium having readable program code ~~embodied in~~ stored on the medium, the computer program product includes:

a first computer program code to create a master view having a master index referencing the hierarchical data;

a second computer program code to create a subordinate view having a subordinate index referencing a subset of said master index, where the subordinate

view defines accessible data of the hierarchical data and the subordinate view index is

linked to a subset of the master view index;

a third computer program code to access the hierarchical data via the

subordinate view; and

a fourth computer program code to create an index map and temporary index for

linking data associated with the subordinate view to the master index.